InnoDB: A hands-on exploration of on-disk storage with innodb_ruby

Jeremy Cole, Davi Arnaut
Friday, April 4, 2014 -- 11:00am to 11:50am

To install

If you have a working RubyGems installation:

To install as root:

```
sudo gem install innodb_ruby
```

To install as a user:

```
gem install --user-install innodb_ruby
```

To run:

```
innodb_space ...
```

If you want to run from git directly:

```
git clone https://github.com/jeremycole/innodb_ruby.git
cd innodb_ruby
```

To run from git:

```
ruby -r rubygems -I lib bin/innodb_space ...
```

Basics of using innodb_space from the command line

There are two ways to start innodb_space.

Against a single space file (ibdata or .ibd):
`-f` = tablespace file name (system or table)

Against a system tablespace which will auto-load file-per-table tablespace files:
`-s` = system tablespace file name
`-T` = table name
`-I` = index name

Useful commands innodb_space from the command line

system-spaces

List all tablespaces available from the system, including some basic stats. This is basically a list of tables:

```
innodb_space -s ibdata1 system-spaces
```

space-indexes

List all indexes available from the space (system space or file-per-table space):

```
innodb_space -s ibdata1 -T sakila/film space-indexes
```

space-page-type-regions

Iterate through all pages in a space and print a summary of page types coalesced into “regions” of same-type pages:

```
innodb_space -s ibdata1 -T sakila/film space-page-type-regions
```

space-page-type-summary

Iterate through all pages and print a summary of total counts of pages by type:

```
innodb_space -s ibdata1 -T sakila/film space-page-type-summary
```

page-account

Given any page number, explain what the page is used for (for most structures):

```
innodb_space -s ibdata1 -T sakila/film -p 3 page-account
```

page-dump

Intelligently dump the contents of a page including a representation of most structures that innodb_ruby understands:

```
innodb_space -s ibdata1 -T sakila/film -p 3 page-dump
```

index-recurse

Recurse an index (perform a full index scan) by following the entire B+Tree (scanning all pages by recursion, not just the leaf pages by list):

```
innodb_space -s ibdata1 -T sakila/film -I PRIMARY index-recurse
```

space-extents
Show the extent descriptor bitmaps (pages marked free or used) for all extents in a space:

```
innodb_space -s ibdata1 -T sakila/film space-extents
```

space-lists
Show a summary of the lists (free, free_frag, full_frag, free_inodes, and full_inodes) for the space, including the list length and the list node information of the first and last pages in the list:

```
innodb_space -s ibdata1 space-lists
```

space-list-iterate
Iterate through all extents in a list and show the extents or inodes in the list:

```
innodb_space -s ibdata1 space-list-iterate -L free_frag
```

space-inodes-summary
Print summary information for each inode in the space:

```
innodb_space -s ibdata1 space-inodes-summary
```

space-extents-illustrate
Illustrate all pages in all extents in the space, showing a colorized block (colored by index/purpose) for each page, sized based on the amount of data in the page:

```
innodb_space -s ibdata1 -T sakila/film space-extents-illustrate
```

space-lsn-age-illustrate
Illustrate all pages in all extents in the space, showing a colorized block (colored by the age of the modification LSN for the page):

```
innodb_space -s ibdata1 -T sakila/payment space-lsn-age-illustrate
```

page-illustrate
Illustrate the content of a page:

```
innodb_space -s ibdata1 -T sakila/film -p 3 page-illustrate
```